GYRO: Analyzing new physics in record time

Mark R. Fahey
Oak Ridge National Laboratory

Jeff Candy
General Atomics

Abstract

GYRO solves the 5-dimensional gyrokinetic-Maxwell equations in shaped plasma geometry, using either a local (*flux-tube*) or global radial domain. It has been ported to a variety of modern MPP platforms including a number of commodity clusters, IBM SPs, SGI Altix and the Cray X1. Because it performs quite well on the X1, the GYRO developers have been able to quickly design and analyze new physics scenarios in record time: (i) transport barrier studies (Phys. Plasmas **11** (2004) 1879), (ii) the local limit of global simulations (Phys. Plasmas **11** (2004) L25), (iii) kinetic electron and finite-beta generalizations of a community-wide benchmark case, (iv) impurity transport with application to fuel separation in burning D-T plasmas (to be submitted to Nuclear Fusion). We will discuss the performance of GYRO across several platforms, and summarize recent physics accomplishments and code optimization work.

1 GYRO Overview

The most promising and aggressively studied concept for power production by fusion reactions is the tokamak. Advances in the understanding and control of tokamak plasmas are continuously being realized, although uncertainties remain in predicting confinement properties and performance of larger reactor-scale devices. The coupled gyrokinetic-Maxwell (GKM) equations provide a foundation for the first-principles calculation of turbulent tokamak transport. For years, the numerical solution of the nonlinear GKM equations has been a computational physics "Grand Challenge".

GYRO is a code that simulates tokamak turbulence by solving the time-dependent, nonlinear gyrokinetic-Maxwell equations for both ions and electrons. It uses a five-dimensional grid and advances the system in time using a second-order, implicit-explicit (IMEX) Runge-Kutta (RK) integrator. GYRO is the only GKM code worldwide that has both *global* and *electromagnetic* operational capabilities. Development has been partially funded by the *Plasma Microturbulence Project*, a fusion SciDAC project. GYRO has been ported to a variety of modern MPP platforms including a number of commodity clusters, IBM SPs and the Cray X1. It has shown good scalability on all these platforms; the extent to which this scalability is enhanced by greater per-processor efficiency will be reported. Recently, the GYRO developers have been able to quickly formulate and analyze new physics scenarios in record time using the Cray X1 at ORNL.

Development of the GYRO core solver began in 1999 and all physics capabilities in the original design specification were realized in 2002. Refinements to the numerical schemes were added after that and concluded by early 2003.

In mid 2003, GYRO was ported and subsequently optimized on the Cray X1 at Oak Ridge National Laboratory (ORNL). GYRO was found to perform quite well on the X1, and thusly the GYRO developers have been able to quickly design and analyze new physics scenarios in record time: (i) transport barrier studies (Phys. Plasmas **11** (2004) 1879), (ii) the local limit of global simulations (Phys. Plasmas **11** (2004) L25), (iii) kinetic electron and finite-beta generalizations of a community-wide benchmark case, (iv) impurity transport with application to fuel separation in burning D-T plasmas (to be submitted to Nuclear Fusion). We will report on the performance of GYRO across several platforms, and also report on the recent physics accomplishments, and code optimization work.

In section 2, we review the GKM equations as implemented in GYRO and follow with an overview of the numerical methods used for their solution. Section 3 focuses on GYRO performance through a comparison of code timings on various architectures. In section 4, recent physics accomplishments

are summarized. Section 5 describes the most recent work on GYRO and what future optimizations are envisioned. Finally, conclusions are given in section 6.

2 GYRO equations

The GKM equations couple the *gyrocenter* distribution, f, to the electromagnetic fields, Φ:

$$\frac{\partial f}{\partial t} = \mathcal{L}_a f + \mathcal{L}_b \Phi + \{f, \Phi\} \tag{1}$$

where

$$\mathcal{F}\Phi = \int\int dv_1\, dv_2\, f$$

and $\mathcal{L}_b$, $\mathcal{L}_b$ and $\mathcal{F}$ are linear operators. The sole nonlinearity, which has a Poisson bracket structure, appears in the gyrokinetic equation. The function $f(\mathbf{r}, v_1, v_2)$ is discretized over a 5-dimensional grid (three spatial and two velocity coordinates), while the 3-dimensional electromagnetic fields $\Phi(\mathbf{r}) = [\phi, A_\parallel]$ are independent of velocity. Here ϕ and $A_\parallel$ are the *electrostatic* and *electromagnetic* potentials, respectively. One averages over the fast orbital motion (gyro-orbit) to eliminate the third velocity-space dimension (gyro-angle). However, this so-called *gyro-averaging* operation introduces *nonlocal* spatial operators ($\mathcal{F}$ above, for example) perpendicular to the magnetic field.

First, a fully spectral decomposition of the fluctuating quantities $(f, \phi, A_\parallel)$ is made,

$$\phi = \sum_n \phi_n(r,\theta)\, e^{-\imath n[\varphi - q(r)\theta]} \ . \tag{2}$$

The integer n labels *toroidal eigenmodes*. Linear studies, still very important for research problems, can be carried out using a single value of n. In Eq. (2), φ is the *toroidal angle*, θ is the *poloidal angle* and q is the *safety factor* (which measures the helicity of the equilibrium magnetic field). Although the physical field, ϕ, is 2π-periodic in θ, the Fourier representation has the implication that the Fourier coefficients, ϕ_n, are nonperiodic, and satisfy the phase condition $\phi_n(r,\pi) = e^{-2\pi\imath n q(r)}\phi_n(r,-\pi)$. Since ϕ is real, the coefficients satisfy $\phi_n^* = \phi_{-n}$. By $f_{\mathrm{s}n}$, we mean the nth toroidal harmonic of f_{s}, or $(f_{\mathrm{s}})_n$. In the present paper, we restrict attention to circular $s-\alpha$ (simple) geometry for brevity. In terms of $f_{\mathrm{s}n}$, the spectral form of the gyrokinetic equation, including collisions, is

$$\frac{\partial f_{\mathrm{s}n}}{\partial t} - \mathcal{C}[f_{\mathrm{s}n} - z_{\mathrm{s}}\alpha_{\mathrm{s}} v_{\parallel\mathrm{s}}(\mathcal{G}A_\parallel)_n] = \mathrm{RHS}_n(r,\theta;\lambda,\epsilon) \tag{3}$$

$$\begin{aligned}\mathrm{RHS}_n = &\left[-\frac{v_{\parallel\mathrm{s}}(r,\theta)}{R_0 q}\frac{\partial}{\partial\theta} + \imath\omega_{\mathrm{ds}}^{(1)} + \imath\omega_{\mathrm{ds}}^{(r)}\frac{\partial}{\partial r}\right] \\ &(h_{\mathrm{s}n} + z_{\mathrm{s}}\alpha_{\mathrm{s}}(\mathcal{G}\mathrm{u}_{\mathrm{s}})_n) \\ &- \imath \mathrm{n}_{\mathrm{s}}\,\omega_{*\mathrm{s}}\,(\mathcal{G}\mathrm{u}_{\mathrm{s}})_n + \imath(q/r)\rho_s\,\{\mathcal{G}\mathrm{u}_{\mathrm{s}}, h\}_n \ ,\end{aligned} \tag{4}$$

where $\mathrm{u}_{\mathrm{s}} \doteq \phi - v_{\parallel\mathrm{s}}A_\parallel$, and

$$v_{\parallel\mathrm{s}}(r,\theta) \doteq \sigma\mu_{\mathrm{s}}\sqrt{2\epsilon T_{\mathrm{s}}(r)\left(1-\lambda\hat{B}(r,\theta)\right)} \ . \tag{5}$$

Above, z_{s} is the *charge*, $\rho_s = c_s/\Omega_{ci}$ is the *ion-sound gyroradius*, $c_s = \sqrt{T_e/m_i}$ is the *sound speed*, $\Omega_{ci} = eB/m_i$ is the *ion cyclotron frequency*, R_0 is the *plasma major radius*, $\hat{B}(r,\theta)$ is the *effective magnetic field strength*, $\mathrm{n}_{\mathrm{s}}(r)$ is the *equilibrium density*, T_{s} is the *equilibrium temperature*, σ is the *sign of the parallel velocity*, and $\alpha_{\mathrm{s}} \doteq \mathrm{n}_{\mathrm{s}}/T_{\mathrm{s}}$. The two velocity coordinates are (ϵ, λ).

The Maxwell equations are written in terms of the $h_{\mathrm{s}n}$ as

$$\sum_{\mathrm{s}=1}^{n_{\mathrm{ion}}} \alpha_{\mathrm{s}} z_{\mathrm{s}}^2\,(1-\mathcal{R}_{\mathrm{s}})\,\phi_n = \sum_{\mathrm{s}=1}^{n_{\mathrm{ion}}+1} z_{\mathrm{s}}\mathrm{V}[\mathcal{G}h_{\mathrm{s}n}] \ , \tag{6}$$

$$\begin{aligned}-\frac{2\rho_s^2}{\beta_e}\nabla_\perp^2 {A_\parallel}_n + &\sum_{\mathrm{s}=1}^{n_{\mathrm{ion}}+1} \alpha_{\mathrm{s}} z_{\mathrm{s}}^2\,\mathrm{V}[v_{\parallel\mathrm{s}}^2 {A_\parallel}_n] \\ &= \sum_{\mathrm{s}=1}^{n_{\mathrm{ion}}+1} z_{\mathrm{s}}\mathrm{V}[v_{\parallel\mathrm{s}}\mathcal{G}h_{\mathrm{s}n}] \ ,\end{aligned} \tag{7}$$

where n_{ion} is the number of ion species (electrons are denoted by $\mathrm{s} = n_{\mathrm{ion}}+1$), and $\mathcal{R}_{\mathrm{s}} = \mathrm{V}[\mathcal{G}\mathcal{G}]$ is a velocity-space-integrated double gyroaverage. The disastrous *Ampère cancellation problem* will occur if one attempts to naively evaluate the term $\mathrm{V}[v_{\parallel\mathrm{s}}^2 {A_\parallel}_n]$ without rewriting the term in an equivalent form that lends itself to a more robust numerical solution. To do electrostatic simulations, one normally sets $A_\parallel = 0$, which is consistent with the limit $\beta_e \to 0$.

The object V above is the 2-dimensional *velocity-integration operator*, defined as

$$\begin{aligned}\mathrm{V}[f_{\mathrm{s}}] \ \doteq \ &\sum_\sigma \frac{1}{2\sqrt{\pi}}\int_0^\infty d\epsilon\, e^{-\epsilon}\sqrt{\epsilon} \\ &\int_0^1 \frac{d(\lambda\hat{B})}{\sqrt{1-\lambda\hat{B}}}\, f_{\mathrm{s}}(r,\theta,\varphi;\epsilon,\lambda,\sigma;t).\end{aligned} \tag{8}$$

For concreteness, we note that $\mathrm{V}[1] = 1$, $\mathrm{V}[\epsilon] = 3/2$ and $\mathrm{V}[v_\parallel] = 0$.

2.1 Discretization schemes

Here we sketch the type of discretization scheme used in each dimension in order to provide a rough idea of the numerical methods used in GYRO. A detailed treatment is beyond the scope of the present paper.

- **toroidal angle:** This is fully spectral [see Eq. (2)].
- **radius:** Linear advective derivatives on f are treated with upwind differences, whereas derivatives on fields are treated with centered differences. The gyrooperators $\mathcal{G}$ and $\mathcal{R}$ are approximated using a (banded) pseudospectral technique. The order of all discretizations is adjustable at run-time.
- **poloidal angle:** For f, there is no fixed grid in θ. Instead, the transformation

$$\frac{v_{\parallel \mathrm{s}}(r,\lambda,\theta)}{R_0 q(r)} \frac{\partial}{\partial \theta} \rightarrow \Omega(r,\lambda) \frac{\partial}{\partial \tau} \qquad (9)$$

  is used to eliminate the singularity at bounce points, $v_{\parallel \mathrm{s}} = 0$. Then, an upwind scheme in τ is used to discretize $\partial f/\partial \tau$. Centered differences only are used on τ-derivatives of fields, otherwise numerical instability will result.
- **velocity-space:** A transformation property of Eq. (8) under integration over θ is used to recast the velocity-space integration. Then, in both ϵ and λ, an exact Gauss-Legendre quadrature scheme is numerically generated (by nonlinear root-finding) at run-time. This is different at each radius and for different plasma equilibria.
- **nonlinearity:** The nonlinear Poisson bracket is evaluated with a *conservative* difference-spectral analogue of the Arakawa method.
- **collisions:** Collisions are represented by a second-order diffusive-type operator in λ. This operator is split from the collisionless problem and a irregular-grid generalization of the Crank-Nicholson method is used.
- **time-advance:** A 2nd-order IMEX RK scheme is used, with the electron parallel motion ($\partial/\partial\theta$) in Eq. (4) treated implicitly. This is exceptionally complicated due to the use of a τ-grid, as well as the presence of the field quantity $\mathrm{u_s}$ in the advection. However, the implicitness is crucial for the elimination of a numerical instability connected with pathological *electrostatic Alfvén waves*.

3 Performance comparisons

In this section we describe the performance of GYRO. As mentioned previously, GYRO has been ported to a variety of modern MPP platforms including a number of commodity clusters, IBM SPs, SGI Altix, and the Cray X1. Since the developers insist on portable code, to date they have only a single source and plan to keep it that way. Thus a port to a new architectures is typically nothing more than creating a new makefile. This is mentioned now because although the single-source philosophy has made GYRO a very portable code, it has the impact of preventing use of machine specific code optimizations which would adversely affect performance on other architectures.

GYRO was ported to the ORNL Cray X1 in mid-2003. In testing functionality, a couple bugs in GYRO were identified and corrected. GYRO has undergone some vectorization for the X1. The changes were minimal – affecting only 14 routines – and in some cases simultaneously beneficial on other architectures. These changes were mostly the addition of directives, but there were selected instances of rank promotion/demotion, and an instance of "pushing" a loop down into a subroutine call.

3.1 Platforms

In following sections, GYRO timings will be reported from the following list of platforms.

- **AMD Athlon cluster**
  The AMD Athlon cluster at Princeton Plasma Physics Laboratory (PPPL) has 48 2-way Athlon MP2000+ (1.667 GHz) nodes with gigE interconnect. Each processor has a peak performance of $1.667 \times 2 = 3.334$ GFlops/s. GYRO was compiled with Lahey/FujitsuFortran 95 L6.10c.

- **IBM Nighthawk II (Power3) cluster**
  The IBM Nighthawk II cluster with SP Switch2 interconnect at the National Energy Research Supercomputer Center (NERSC) has 416 16-way Power3 (375 Mhz) nodes. The peak performance of a Power3 processor is $375 \times 4 = 1.5$ GFlops/s. GYRO was compiled using ESSL 3.3 and XL Fortran 8.1. The machine has parallel environment 3.2 and is running AIX 5.1.
- **IBM p690 (Power4) cluster**
  The IBM p690 cluster with Federation interconnect[1] at Oak Ridge National Laboratory (ORNL) has 27 32-way Power4 (1.3 GHz) nodes. The peak rate of each node is $32 \times 1.3 \times 4 = 166.4$ GFlops/s. GYRO was built using ESSL 4.1 and XL Fortran 8.1. The machine is using parallel environment 4.1 and running AIX 5.2.
- **SGI Altix**
  The SGI Altix at ORNL is a 256-way single-system image with a NUMAflex fat-tree interconnect running Linux. Each processor is a 1.5 GHz Itanium 2 with 6 MB L3 cache. The machine peak rate is $256 \times 1.5 \times 4 = 1536$ GFlops/s. The Altix has a NUMAflex fat-tree interconnect. GYRO was built using the Intel Fortran 7.1 compiler.
- **Cray X1**
  The Cray X1 at ORNL is also a single-system image of 256 multistreaming processors. Each processor is capable of 12.8 GFlops/s, and thus the peak of the machine is 3276.8 GFlops/s. The OS and Programming Environment have changed constantly on the X1 at ORNL over the past year as the software matured. Thus there is no one OS level or version of the compiler that all the tests were run at. Note the latest GYRO timings were collected with Programming Environment 5.1.0.5 and mpt 2.3.0.3, and the OS was Unicos m/p 2.4.12. Unless otherwise stated, all X1 timings are reported MSP counts.

[1] Currently the Federation interconnect does not properly stripe across adapters with 2 links. In this case where the number of parallel communication paths made available to the network protocol is 2, the performance is worse than if the number of paths is set to 1. Thus `max_proto_instances=1` on the ORNL p690 cluster as recommended by IBM until a "Federation code with striping" is released.

3.2 BCY.n16.25

This test case is a 16-toroidal-mode electromagnetic (electrons and ions, 2 fields) case. The grid is $16 \times 128 \times 8 \times 8 \times 20 \times 2$. It runs on multiples of 16 processors. The test case run for 8 simulation seconds representing 1000 timesteps. With an 128-point radial domain and only 16 toroidal modes, this benchmark is somewhat small. However, the electromagnetic physics (field-line bending) requires small timesteps for nonlinear stability.

Figure 1 shows the average number of seconds per timestep on three platforms for BCY.n16.25, and Figure 2 shows the ratio of communication time to total time on these same machines.

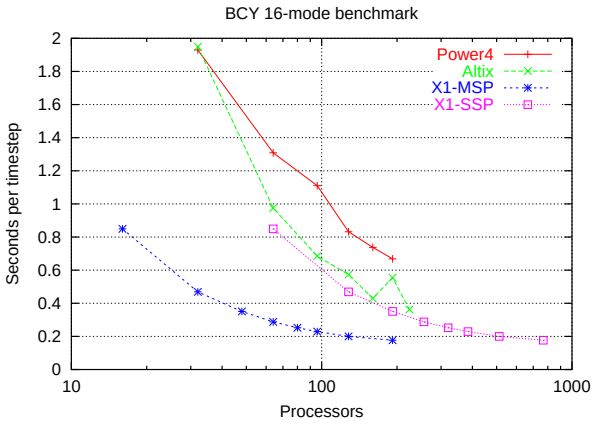

Figure 1: Average number seconds per timestep.

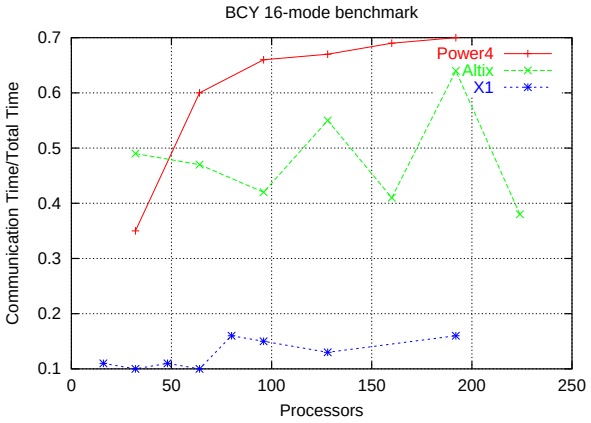

Figure 2: Communication Time / Total Time.

Figure 1 shows that X1 performance is significantly better than that on other platforms, even

for this modest size problem. Even replotting the data with Single-Streaming Processors (SSPs), four of which compose one MSP, as the processor indicates that the X1 is the faster platform. Furthermore, Figure 2 shows that the IBM performance is limited by communication overhead. The large spike in the communication time percentage at 128 and 192 processors on the Altix is not understood yet, and it is reproducible. Also, although not obvious, Figure 2 shows that X1 performance is limited by a nonscaling part of the parallel algorithm. At 192 processors, 15% of the time is spent in a phase that does not scale beyond 16-way parallelism.

3.3 GTC.n64.500a

These results are for a larger case GTC.n64.500a which is a 64-toroidal-mode adiabatic electron (ions only, 1 field) case. The grid is $64 \times 400 \times 8 \times 8 \times 20 \times 1$. This runs on multiples of 64 processors. The duration is 3 simulation seconds, representing 100 timesteps. The 400-point radial domain with 64 toroidal modes gives extremely high spatial resolution, but electron physics is ignored allowing simple field solve and large timestep. The primary communication cost results from calls to `MPI_ALLTOALL`. These are required to transpose distributed arrays (distribution functions).

Figure 3 shows the average number of seconds per timestep on three platforms for GTC.n64,500a, and Figure 4 shows the ratio of communication time to total time on these same machines.

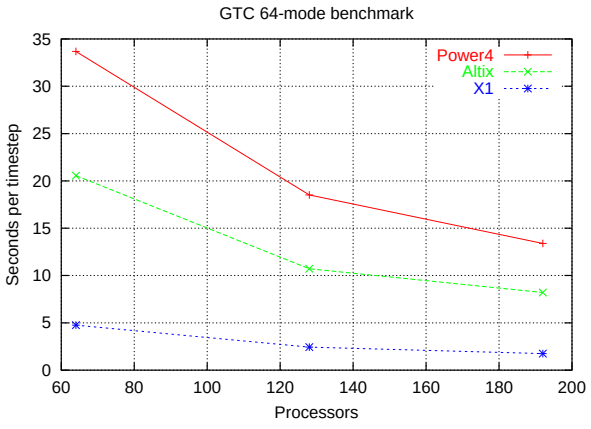

Figure 3: Average Number of seconds per timestep.

Figure 3 shows the X1 with a strong performance here as in the previous 16-mode case. For this 64-mode case, the X1 performance is about a factor of 4 better than the Altix and a factor of 7 better than the IBM. Figure 4 shows that both the IBM and SGI are limited by communication overhead. Note that the X1 performance is again affected by the nonscaling part of parallel algorithm (as for the 16-mode case), but this time the affect is negligible. For instance, at 192 processors, 3% of the time is spent in a phase that does not scale beyond 64-way parallelism. This effect is minor because at 192 processors, this case is using 1/3 of the processor while for the 16-mode case it is only using 1/12 of the processors. The communication is at least of factor of 5 better than the nearest competitor.

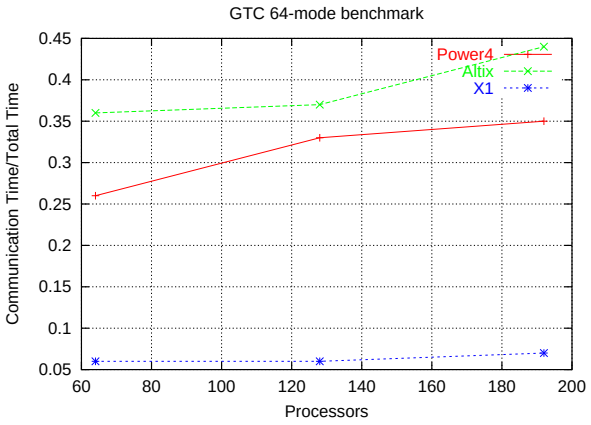

Figure 4: Communication Time / Total Time.

3.4 Waltz standard case benchmark

In this section, results are presented for the Waltz standard case benchmark. This is a 16-toroidal-mode electrostatic (electrons and ions, 1 field) case on a $16 \times 140 \times 8 \times 8 \times 20 \times 2$ grid. This runs on multiples of 16 processors, and is similar to the BCY.n16.25 benchmark. This test case is run for 500 timesteps with kinetic electrons and electron collisions, but no electromagnetic effects.

The numerical grid resolution used for this benchmark is the same as that used in production runs. It represents, roughly, the *minimum grid size* required to obtain physically accurate results. The scalability on all platforms could have been improved by utilizing a finer grid, but is of more practical value since a production-sized grid was used.

This test case demonstrates that the X1 yields poor performance on "collisions" in GYRO. How-

ever, the X1 is still the fastest architecture.

Figures 5-7 show the results of running the Waltz standard case benchmark on the various platforms mentioned above. Figure 5 shows the seconds per timestep - the ultimate measure of how well this code does on any machine. Figure 6 shows the average MPI time per timestep; this is derived from total communication time and some extra integer algebra. Figure 7 shows the average collision time per timestep.

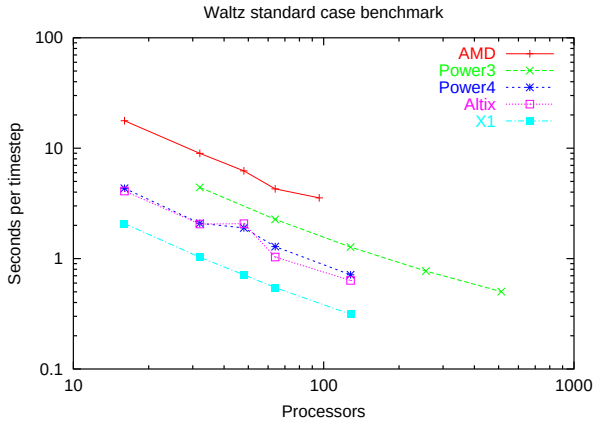

Figure 5: Seconds per timestep on the Waltz standard case benchmark.

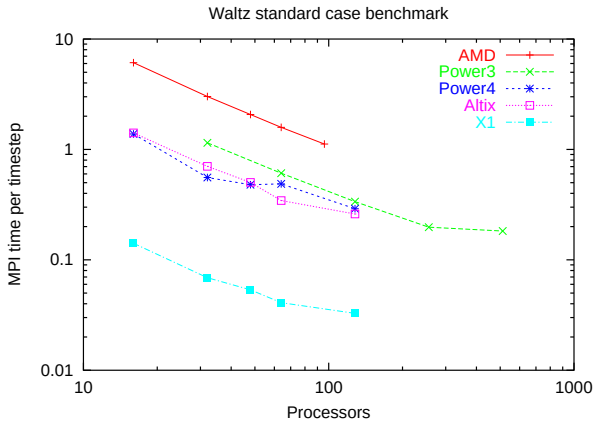

Figure 6: MPI time (s) per timestep on the Waltz standard case benchmark.

A few observations can be made from Figures 5-7. First, it is clear from Figure 6 that the X1 provides more bandwidth than the the other platforms with an order of magnitude less time spent in MPI. Second, Figure 7 also makes it clear that the collision step runs several times slower on the X1 than the other platforms and thus requires further tuning. From Figure 5 we see for this benchmark that the X1 is twice as fast as the nearest competitor. If we ignore the collisions cost (there are real simulations where the collision step is not used), we see that the X1 is 5 times faster than the nearest competitor. We also see from Figures 5 and 6 that the IBM Power4 cluster with Federation interconnect compares favorably with the other machines and that it scales fairly well even with its current handicap (see footnote 1.) Note that the AMD cluster also performs reasonably well only being an order of magnitude slower than the fastest machine; and although its interconnect is slow it scales well too.

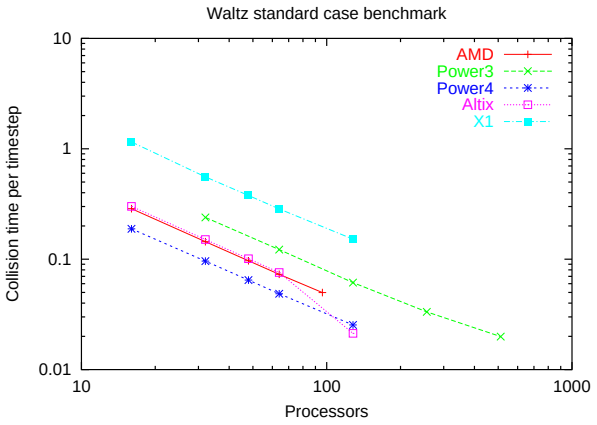

Figure 7: Collision time (s) per timestep on the Waltz standard case benchmark.

3.5 Exploratory plasma edge simulation

Also very recently we had the opportunity to run a very large test case on 504 MSPs of a Cray X1. This time was used to run a gyrokinetic simulation meant to explore the parameter space characteristic of the plasma edge. Magnetic shear and safety factor are very high at the outer plasma boundary, and equilibrium gradients are steep, making the simulations more challenging than for core parameters.

A visualization of the results from a 504 MSP simulation on the Cray X1 is shown in Figure 8. This plots contours of the turbulent electrostatic potential mapped back onto an (elongated) torus. Note the extreme elongation of the turbulent eddies in the toroidal direction.

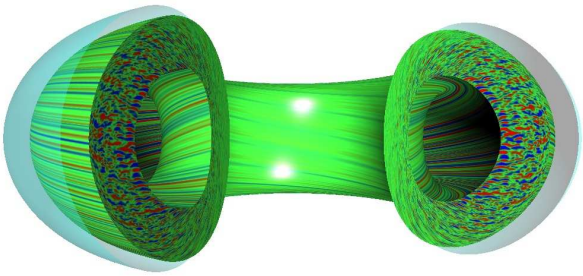

Figure 8: Turbulent potential fluctuations in shaped toroidal geometry for plasma edge simulation. This case was run on 504 MSPs of the Cray X1.

Table 1 compares the performance of running this exploratory simulation on an IBM Power3 cluster and on a Cray X1 at relatively large processor counts. The "time/step" column shows the average time per step for 200 time steps of this simulation, and the "MPI/step" column shows the average time per step spent in MPI. Note that this is a collisionless simulation.

Table 1: Exploratory plasma edge simulation timings.

Machine	procs	time/step	MPI/step
IBM Pwr3	896	0.602450	0.103694
cluster	1344	0.544581	0.081436
	1792	0.405187	0.067532
	2240	0.431481	0.073186
	2688	0.422913	0.066386
Cray X1	504	0.072615	0.005889

This study has shown that the Cray X1 at 504 MSPs can take almost 13.8 steps per second, whereas the same case on the IBM NERSC Power3 machine reaches a maximum rate of 2.5 steps per second at 1792 PEs and beyond. This means in practice that one can take 5.8 times more steps per second on the X1 (with 3.5 times fewer processors) than on the IBM. The greater per-processor efficiency is clearly evident with the X1 and leads to greater scalability.

4 Physics Accomplishments

The following detail some of the more important findings made in the last year using the X1.

4.1 Comparison with DIII-D L-mode ρ_* experiments

An exhaustive series of global, full-physics GYRO simulations of DIII-D L-mode ρ_*-similarity discharges was made. In these famous experiments, only a single dimensionless parameter was varied: $\rho_* \doteq \rho_s/a$. The GYRO calculations matched experimental results for electron and ion energy transport [1] within experimental error bounds. The Bohm-scaled diffusivity of the experiments was also reproduced, for which inclusion of electron collisions together with equilibrium sheared rotation was found to be crucial. These were the most physically comprehensive tokamak turbulence simulations ever undertaken.

4.2 Evaluation of minimum-q theory of transport barrier formation

It was shown that a minimum-q surface (where $s = 0$) in a tokamak plasma does not act as the catalyst for ion transport barrier formation [3]. Although there are theories of this process in the literature which argue that a barrier should form near $s = 0$, the reality of this process had never been convincingly demonstrated with simulation. Using the X1, it was clearly shown that transport is smooth across an $s = 0$ surface due to the appearance of *gap modes*.

4.3 The local limit of global GK simulations

A controversial transport scaling study by Lin *et al.* [5] substantially overestimated the *Cyclone base case* [4] benchmark value as ever-larger global simulations (at successively smaller ρ_*) were run. This contradicted the *local hypothesis*, which states that global and flux-tube simulations should agree at sufficiently small ρ_*. Lin repeated the same study with minor revisions in 2002 [7] and yet again in 2004 [6] obtaining an ion diffusivity, χ_i, still 36% higher than the Cyclone value. Lin's global scenario was revisited with GYRO [2], with the finding that at small ρ_*, χ_i closely matches the Cyclone value. Further, it was shown that for these large-system-size simulations, there is a very long transient period for which χ_i exceeds the statistical average. By running simulations for twice as long as the Lin simulations (900 a/c_s versus 400 a/c_s), a true statistical turbulent steady state was recovered.

4.4 Particle and impurity transport

The first systematic gyrokinetic study of particle transport was made, including impurity transport and isotope effects. We found that in a burning D-T plasma, the tritium is better confined than deuterium, with the implication that the D-T fuel will separate as tritium is retained. This conclusion was found to be independent of temperature gradient and electron collision frequency.

5 Current and future work

In this section we discuss two recent issues with GYRO: 1) numerical accuracy investigation of GYRO on the X1 and 2) the performance of the collision step on the X1. And we conclude with a discussion of the optimization work planned for GYRO.

5.1 Catastrophic cancellation

In late 2003, simulation results from the X1 were found to agree to only 9 digits from results on other architectures after the setup phase. The timestepping portion of the code had not even started. The developers were of course quite concerned about this and so an investigation was undertaken to understand why. After a couple months of searching and hundreds of print statements, one numerical issue (in two separate routines) was found to be the cause of the problem - *catastrophic cancellation*. Catastrophic cancellation is the sudden loss of accuracy usually due to subtracting nearly equal quantities.

The GYRO numerical grid contains *bounce points* - points where $v_\parallel$ changes sign. Particles for which $v_\parallel$ change sign are called *trapped*. The code computes $v_\parallel = \sqrt{1 - \lambda B(\theta)}$ where (approximately) $B = 1 - \epsilon\cos(\theta)$ and $\epsilon = r/\rho$ [see Eq. (5)]. Thus, at a bounce point, $v_\parallel$ is identically zero. However, numerically the code generates values $\mathcal{O}(\sqrt{\varepsilon})$ because $\lambda B(\theta_b)$ is nearly equal to 1 where θ_b is the solution of $B(\theta) = 1/\lambda$. Thus, at bounce points, where the exact result should have been zero, we put in exceptional cases to set the result to zero.

With the exceptional cases implemented in the code for the bounce points, now the results match other platforms to 12 digits after the setup phase. More digits of agreement is not expected or needed.

This loss of accuracy originating at the bounce points (which is 1/40th of all gridpoints in a typical simulation) was roughly equivalent to adding a stochastic source term with amplitude 1e-9 to the $\mathcal{O}(1)$ kinetic equations. This can be shown that it makes little difference in the "time-averaged" turbulent diffusivity, which is an $\mathcal{O}(1)$ quantity. Thus, the physical interpretation of all results prior to adding the exceptional cases for the bounce points are valid. And now GYRO is more arithmetically robust.

5.2 Collisions

The collision step on the X1 is performs poorly as exhibited by the Waltz standard case benchmark above. Figure 7 showed that the X1 does the collision step many times slower than competing architectures. Although the collision step has already undergone optimizations to improve vectorization, the results clearly show that more work is necessary.

To date the optimization attempts on the collision step have concentrated on the collision routine solely, and have not significantly changed the underlying algorithm, the calling routine, or the data structures. For example, the first "optimizations" inlined the tridiagonal solver and removed the pivoting logic. This resulted in improved vectorization, but a tridiagonal solve is still a serial operation at its core. The second attempt was to vectorize across the tridiagonal solves. For most simulations, there are 64 or more independent tridiagonal solves. So the algorithm was modified to concurrently solve these systems. The hardest part of the logic to implement was how to concurrently solve 64 tridiagonal systems whose order is not guaranteed to be the same. After implementing this second optimization, the tridiagonal matrix "setup" in the collision routine is the most time-consuming step.

Currently, we believe the collision routine considered stand-alone from the rest of the code has been optimized as much as possible for the X1. To truly fix the poor performance, we expect that a major rewrite of the collision routine will be required.

5.3 Future work

The future of GYRO with respect to optimizations is clear: first and foremost the collision step must be rewritten so that it performs much better on the X1. Secondarily, a new parallel algorithm for the field solves which currently replicates work must be implemented. This inhibits scalability in some cases, and will be beneficial on all platforms. After these first two obvious candidates, we plan to improve the

nonlinear step by evaluating the transformation of the toroidal angle in real space. This will mean a more efficient algorithm using FFTs and again will benefit all platforms. Lastly, a review of alternatives to the current sparse solver will be done.

6 Conclusions

The X1 has provided a platform where new physics scenarios have been quickly designed and analyzed in record time in the last year. These findings include comparison with DIII-D L-mode ρ_* experiments, evaluation of minimum-q theory of transport barrier formation, simulations of the local limit of global GK simulations, and a systematic study of particle and impurity transport.

The performance of GYRO on nonvector systems is constrained by communication bandwidth, which is not true on the X1. The X1 has exceptionally high-bandwidth, low-latency communication hardware, so communication times for various expensive ALL TO ALL operations dropped to the lowest levels ever for GYRO (the flagship gyrokinetic code for the US Fusion Program). Moreover, the powerful vector processing units each give a performance that is a factor of 20 greater than Power3 for some cases like the exploratory edge plasma simulation. The fast communication coupled with the intrinsically lower processor count define a system which, for a given production problem size, can in many cases provide a time-to-solution which is smaller than a traditional supercomputer of any size.

The collision step on the X1 is known to perform poorly, and the Waltz standard case benchmark is a perfect example of that. The next step in the optimization process is to address this issue by re-evaluating the collision algorithm and to implement something more vector friendly. However, even for problems that do "collisions", the X1 is still at least a factor of two better than its competition. And for collisionless simulations, the advantage is significantly more.

Acknowledgments

Thanks to Pat Worley for granting us permission to use the BCY and GTC benchmark data he gathered.

About the Authors

Mark R. Fahey is in the Scientific Application Analyst, Center for Computational Sciences (CCS) at Oak Ridge National Laboratory, and is the primary CCS liaison for fusion researchers funded by the DOE. He is the current CUG Application SIG chair. He can be reached at Oak Ridge National Laboratory, P.O. Box 2008 MS6008, Oak Ridge, TN 37831-6008, E-Mail: faheymr@ornl.gov.

Jeff Candy is a staff scientist at General Atomics. He received an M.Sc. (theoretical physics) from the University of Alberta, and a Ph.D. (physics) from UCSD. His current research interests include linear and nonlinear kinetic theory, numerical analysis, parallel algorithms, and gyrokinetic simulation. He can be reached at General Atomics, P.O. Box 85608, San Diego, CA 92186-5608, E-Mail: candy@fusion.gat.com.

References

[1] J. Candy and R.E. Waltz. Anomalous transport in the DIII-D tokamak matched by supercomputer simulation. *Phys. Rev. Lett.*, 91:045001–1, 2003.

[2] J. Candy, R.E. Waltz, and W. Dorland. The local limit of global gyrokinetic simulations. *Phys. Plasmas*, 11:L25, 2004.

[3] J. Candy, R.E. Waltz, and M.N. Rosenbluth. Smoothness of turbulent transport across a minimum-q surface. *Phys. Plasmas*, 11:1879, 2004.

[4] A.M. Dimits, G. Bateman, M.A. Beer, B.I. Cohen, W. Dorland, G.W. Hammett, C. Kim, J.E. Kinsey, M. Kotschenreuther, A.H. Kritz, L.L. Lao, J. Mandrekas, W.M. Nevins, S.E. Parker, A.J. Redd, D.E. Shumaker, R. Sydora, and J. Weiland. Comparisons and physics basis of tokamak transport models and turbulence simulations. *Phys. Plasmas*, 7:969, 2000.

[5] Z. Lin, S. Ethier, T.S. Hahm, and W.M. Tang. Size scaling of turbulent transport in mag-

netically confined plasmas. *Phys. Rev. Lett.*, 88:195004, 2002.

[6] Z. Lin and T.S. Hahm. Turbulence spreading and transport scaling in global gyrokinetic particle simulations. *Phys. Plasmas*, 11:1099, 2004.

[7] Z. Lin, T.S. Hahm, S. Ethier, W.W. Lee, J. Lewandowski, G. Rewoldt, W.M. Tang, W.X. Wang, L. Chen, and P.H. Diamond. Size scaling of turbulent transport in tokamak plasmas. In *Plasma Physics and Controlled Nuclear Fusion Research (Proc. 19th International Conference on Fusion Energy, Lyon, 2002)*. IAEA, Vienna, 2003.